



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|-------------|----------------------|---------------------|------------------|
| 10/087,723 | 03/01/2002 | Tony Gargya | DE920010003US1 | 6694 |
| 7590 | 02/04/2005 | | EXAMINER | |
| Jerry W. Herndon IBM Corporation T81/503 PO Box 12195 Research Triangle Park, NC 27709 | | | MOFIZ, APU M | |
| | | | ART UNIT | PAPER NUMBER |
| | | | 2165 | |

DATE MAILED: 02/04/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

10/087,723

Applicant(s)

GARGYA ET AL.

Examiner

Apu M Mofiz

Art Unit

2165

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 01 September 2004.
2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-6 and 8-17 is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5) ☐ Claim(s) _____ is/are allowed.
6) ☒ Claim(s) 1-6, 8-11, 13 and 15-17 is/are rejected.
7) ☒ Claim(s) 12 and 14 is/are objected to.
8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
10) ☒ The drawing(s) filed on 01 March 2002 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

Response to Applicant's Remarks

1. In response to applicant's remarks, all previously presented rejections of claims are hereby withdrawn as to being moot.

Claim Rejections - 35 USC § 102

2. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

3. Claims 1-6, 8-10, 13 and 15-17 are rejected under 35 U.S.C. 102(a) as being anticipated by Cape Clear (CapeConnect 1.0 Technical Overview, February 2001 and CapeClear hereinafter).

As to claims 1,13 and 5-7, CapeClear teaches a method for coupling a client (4) (i.e. "At run time, information from a Web or SOAPDirect client is channeled to CapeConnect, which interprets the client's input and converts it to CORBA calls.") (page 2; page 3) of a first object type (i.e. "Using SOAPDirect, you can integrate any Java-based program with back-end CORBA systems through the CapeConnect XML infrastructure.") (page 3; page 14) to a server (8, 9) of a second object type (i.e. "CapeConnect 1.0 makes CORBA objects available to clients across the Internet or an Intranet." ... "These elements connect server-side components, such as CORBA objects, to a universal,

Art Unit: 2165

Web-based infrastructure." ... "SOAPDirect, in conjunction with the CapeConnect servlet and XML engine, provides a HTTP-based transport for communicating with CORBA objects through XML." (page 3; page 14) via a bridge (i.e. "CapeConnect is an Extensible Markup Language (XML) business server that connects back-end computer systems to a semantically rich, Web-based infrastructure." ... "CapeConnect uses XML to bridge the gap between the Internet and back-end systems." ... "The CapeConnect servlet runs in a servlet engine on your Web server. It acts as a communication bridge between remote clients and the CapeConnect XML engine.") (page 1; page 5), the bridge (page 1; page 5) comprising a server component (11) of the first object type (page 3; page 14), a client component (13) of the second object type (page 3; page 14) and a mapping component (17) for mapping (i.e. "At run time, information from a Web or SOAPDirect client is channeled to CapeConnect, which interprets the client's input and converts it to CORBA calls. CapeConnect communicates with your CORBA objects, gathers the information that a client requires, and converts this information to a format suitable for client to process or a user to view." ... "CapeConnect maps CORBA calls to corresponding Web communications.") (page 2; page 11) of objects of the first object type (page 3; page 14) to corresponding objects of the second object type (page 3; page 14), the method comprising the steps of - coupling (page 3; Figure 1) the client to the bridge (page 1; page 5), - providing a uniform resource locator (i.e.

"Public static void main (String[] args) {

String XMLServletURL = http://localhost:8080/servlet/XMLServlet;

...

...

SDRequest helloRequest = new SDRequest (XMLServletURL, "SimpleServer", "SimpleApp",

"Simple", "hello");" the previous code text excerpts clearly indicate that a SOAPDirect Client of first object type "Java" provides a URL of the bridge (i.e., CapeConnect Servlet) and the CORBA object name, the IDL interface name, and the IDL operation name in the method parameters to the Web Server or the CapeConnect Servlet.) (page 3; page 15; page 16) of the server to the bridge (page 3; page 15;

Art Unit: 2165

page 16) and providing parameter data to the bridge (page 1; page 5) indicative of the object type (i.e. CORBA object type) (page 3; page 15; page 16) of the server (page 3; page 15; page 16), -coupling the bridge to the server via the client component of the bridge having the same object type (i.e. the CapeConnect XML engine sends CORBA request over IIOP to the CORBA Server.) (page 3; page 15; page 16) as the server (page 3; page 15; page 16).

As to claim 2, CapeClear teaches that wherein the first object type (i.e. *"Using SOAPDirect, you can integrate any Java-based program with back-end CORBA systems through the CapeConnect XML infrastructure."*) (page 3; page 14) uses the simple object access protocol (SOAP) type (i.e. *"Similarly, all CapeConnect XML communications comply with the Simple Object Access Protocol (SOAP) 1.1 specification."*) (page 1; page 3).

As to claim 3, CapeClear teaches wherein the second object type (i.e. *"CapeConnect 1.0 makes CORBA objects available to clients across the Internet or an Intranet."* ... *"These elements connect server-side components, such as CORBA objects, to a universal, Web-based infrastructure."* ... *"SOAPDirect, in conjunction with the CapeConnect servlet and XML engine, provides a HTTP-based transport for communicating with CORBA objects through XML."*) (page 3; page 14) is the enterprise Java Beans (EJB) type or the common object request broker architecture (CORBA) type (i.e. *"CapeConnect 1.0 makes CORBA objects available to clients across the Internet or an Intranet. (The next release of CapeConnect will add support for EJBs.)"*) (page 3).

As to claim 4, CapeClear teaches wherein the bridge (i.e. *"CapeConnect is an Extensible Markup Language (XML) business server that connects back-end computer systems to a*

Art Unit: 2165

semantically rich, Web-based infrastructure.” ... *“CapeConnect uses XML to bridge the gap between the Internet and back-end systems.”* ... *“The CapeConnect servlet runs in a servlet engine on your Web server. It acts as a communication bridge between remote clients and the CapeConnect XML engine.”*) (page 1; page 5) has means for coupling to the client (i.e. *“At run time, information from a Web or SOAPDirect client is channeled to CapeConnect, which interprets the client’s input and converts it to CORBA calls.”*) (page 2; page 3) of the first object type (i.e. *“Using SOAPDirect, you can integrate any Java-based program with back-end CORBA systems through the CapeConnect XML infrastructure.”*) (page 3; page 14) via the internet (i.e. *“CapeConnect 1.0 makes CORBA objects available to clients across the Internet or an Intranet.”*) (page 3; page 14).

As to claim 5, CapeClear teaches wherein the bridge (i.e. *“CapeConnect is an Extensible Markup Language (XML) business server that connects back-end computer systems to a semantically rich, Web-based infrastructure.”* ... *“CapeConnect uses XML to bridge the gap between the Internet and back-end systems.”* ... *“The CapeConnect servlet runs in a servlet engine on your Web server. It acts as a communication bridge between remote clients and the CapeConnect XML engine.”*) (page 1; page 5) has means for coupling to the server (i.e. *“CapeConnect 1.0 makes CORBA objects available to clients across the Internet or an Intranet.”* ... *“These elements connect server-side components, such as CORBA objects, to a universal, Web-based infrastructure.”* ... *“SOAPDirect, in conjunction with the CapeConnect servlet and XML engine, provides a HTTP-based transport for communicating with CORBA objects through XML.”*) (page 3; page 14) via an intranet (page 3; page 14).

As to claim 6, CapeClear teaches wherein the bridge (i.e. *“CapeConnect is an Extensible Markup Language (XML) business server that connects back-end computer systems to a semantically rich, Web-based infrastructure.”* ... *“CapeConnect uses XML to bridge the gap between the*

Art Unit: 2165

Internet and back-end systems.” ... “The CapeConnect servlet runs in a servlet engine on your Web server. It acts as a communication bridge between remote clients and the CapeConnect XML engine.” (page 1; page 5) has an assigned uniform resource locator (URL) (i.e. *“A superset of URL, and a necessary part of the HTTP headers in a SOAP request.”*) (page 27) for access of the bridge (page 1; page 5) from the client (i.e. *“At run time, information from a Web or SOAPDirect client is channeled to CapeConnect, which interprets the client’s input and converts it to CORBA calls.”*) (page 2; page 3).

As to claim 8, CapeClear teaches a receiving component (i.e. *“At run time, information from a Web or SOAPDirect client is channeled to CapeConnect, which interprets the client’s input and converts it to CORBA calls. CapeConnect communicates with your CORBA objects, gathers the information that a client requires, and converts this information to a format suitable for client to process or a user to view.”*) (page 2; page 11) for application specific parameters (i.e. *“When making a call to a CORBA object, a client usually sends input parameter values and then waits for a corresponding set of results. For example, a call that calculates the sum of two integers would take the two integer values as input and return a single integer value as a result.”*) (page 10) to be provided to the server (i.e. *“CapeConnect 1.0 makes CORBA objects available to clients across the Internet or an Intranet.” ... “These elements connect server-side components, such as CORBA objects, to a universal, Web-based infrastructure.” ... “SOAPDirect, in conjunction with the CapeConnect servlet and XML engine, provides a HTTP-based transport for communicating with CORBA objects through XML.”*) (page 3; page 14) as input data (page 10).

As to claim 9, CapeClear teaches a client (i.e. *“At run time, information from a Web or SOAPDirect client is channeled to CapeConnect, which interprets the client’s input and converts it to CORBA calls.”*) (page 2; page 3) computer of a first object type (i.e. *“Using SOAPDirect, you*

Art Unit: 2165

can integrate any Java-based program with back-end CORBA systems through the CapeConnect XML infrastructure.") (page 3; page 14) **and a bridge** (i.e. *"CapeConnect is an Extensible Markup Language (XML) business server that connects back-end computer systems to a semantically rich, Web-based infrastructure."* ... *"CapeConnect uses XML to bridge the gap between the Internet and back-end systems."* ... *"The CapeConnect servlet runs in a servlet engine on your Web server. It acts as a communication bridge between remote clients and the CapeConnect XML engine."*) (page 1; page 5) **for establishing a communication** (page 1; page 5) **path between the client computer** (page 2; page 3) **and a server computer** (i.e. *"CapeConnect 1.0 makes CORBA objects available to clients across the Internet or an Intranet."* ... *"These elements connect server-side components, such as CORBA objects, to a universal, Web-based infrastructure."* ... *"SOAPDirect, in conjunction with the CapeConnect servlet and XML engine, provides a HTTP-based transport for communicating with CORBA objects through XML."*) (page 3; page 14) **of a second object type** (i.e. *"CapeConnect 1.0 makes CORBA objects available to clients across the Internet or an Intranet."* ... *"These elements connect server-side components, such as CORBA objects, to a universal, Web-based infrastructure."* ... *"SOAPDirect, in conjunction with the CapeConnect servlet and XML engine, provides a HTTP-based transport for communicating with CORBA objects through XML."*) (page 3; page 14).

As to claim 10, CapeClear teaches a firewall component (18) (i.e.

"Communications between clients, the servlet, and the XML engine take place over HTTP, so you can place firewalls between these components without opening additional, insecure pathways.") (page 3; page 5) **in the communication path** (page 1, page 5).

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

Art Unit: 2165

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

5. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over Cape Clear (CapeConnect 1.0 Technical Overview, February 2001 and CapeClear hereinafter) in view of Larry Peterson ("A Yellow-Pages Service for a Local-Area Network", Department of Computer Science, University of Arizona, 1988, ACM and Peterson hereinafter).

The teachings of CapeClear have been discussed above.

As to claim 11, CapeClear does not explicitly teach wherein the client computer has means for accessing a yellow pages type server for querying the yellow pages type server to obtain a uniform resource locator of a desired location of the server computer.

Peterson teaches that the client computer (i.e. *"Clients specify the attributes the server should possess when requesting a service and the yellow-pages service determines what servers satisfy the request."*) (page 235) has means for accessing a yellow pages type server (i.e. *"A set of servers implement the yellow-pages service."*) (page 235) for querying (i.e. *"We introduce a yellow-pages service that maps service names into server addresses." ... "A client that wishes to engage a service queries the yellow-pages service for the address of a server, specifying the name of the service and any attributes the server should possess."*) (page 235) the yellow pages type server (page 235) to obtain a uniform resource locator of a desired location (i.e. *"We introduce a yellow-pages service that maps service names into server addresses." ... "In addition to describing the implementation of the yellow-pages service within a local area network, we show how the service can be integrated with*

Art Unit: 2165

the available internet communication protocols to enable clients from throughout the internet to access local servers." ... The yellow-pages service returns the address of one or more servers that satisfy the client's requirements." The preceding text indicates that the client queries for the yellow-pages server for the address of the intended service provider server. In the Internet environment the address is an URL.) (page 235) of the server computer (page 235).

It would have been obvious to a person of ordinary skill in the art at the time of Applicant's invention to modify the teachings of CapeClear with the teachings of Peterson to include that the client computer has means for accessing a yellow pages type server for querying the yellow pages type server to obtain a uniform resource locator of a desired location of the server computer with the motivation to connect to the server that is willing to provide the service the client needs (Peterson, page 235) and because it is cost effective (Peterson, page 241).

Allowable Subject Matter

6. Claims 12 and 14 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

The following is a statement of reasons for the indication of allowable subject matter:

As to claims 12 and 14, Cape Clear (CapeConnect 1.0 Technical Overview, February 2001 and CapeClear hereinafter), and Larry Peterson ("A Yellow-Pages Service for a Local-Area Network", Department of Computer Science, 1988, ACM and Peterson hereinafter) do not disclose, teach or suggest the claimed limitations of (in combination with all other features in the claims), the

Art Unit: 2165

computer system wherein the yellow page server provides the object type of the application and, in case of different server and client object types, the uniform resource locator of the bridge.

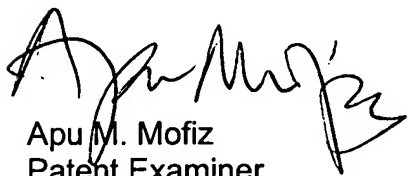
The closest prior arts fail to anticipate or render Applicant's limitations above obvious.

Points of Contact

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Apu M. Mofiz whose telephone number is (571) 272-4080. The examiner can normally be reached on Monday – Thursday 8:00 A.M. to 4:30 P.M.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Dov Popovici can be reached at (571) 272-4083. The fax numbers for the group is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application should be directed to the Group receptionist whose telephone number is (703) 305-9600.



Apu M. Mofiz
Patent Examiner
Technology Center 2100

January 31, 2005